AUTOMATED TESTING FRAMEWORK

By Julio M. Merino Vidal <u>jmmv@NetBSD.org</u> NYCBSDCon 2008

What? No BSD in the title?

Major NetBSD goals

- 3
- □ Despite what popular culture says...
 - Correction
 - Stability
- Open questions:
 - How to quantify these features?
 - How to verify these features?
 - When to evaluate these features?

Automated tests!

Contents

- 5
- □ The old NetBSD tests
- NetBSD needs for a testing framework
- □ Brief introduction to ATF
- The ATF libraries and tools
- □ The new NetBSD tests

The old NetBSD tests

- Located in src/regress
- □ How are they?
 - Little, independent programs (sometimes)
 - Some code in Makefiles
- □ But... do they really have problems?
 - Interface inconsistency
 - Difficulty in automating their execution
 - Need the source code to run them

What did NetBSD want/need?

- Single, unified results report
- □ Interface consistency
- □ Isolation and remote/distributed execution
- □ No need for developer-only tools (aka comp.tgz)
- Automation
- □ Automation

And here comes ATF...

What is ATF?

Automated Testing Framework

What is ATF?

□ A framework, not a set of tests

Goals

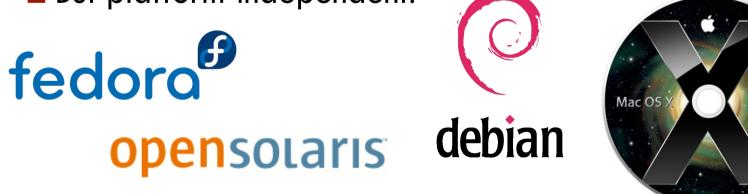
10

Ease writing of tests

Ease execution of tests

Project under the The NetBSD Foundation umbrella

But platform-independent!



Leopard

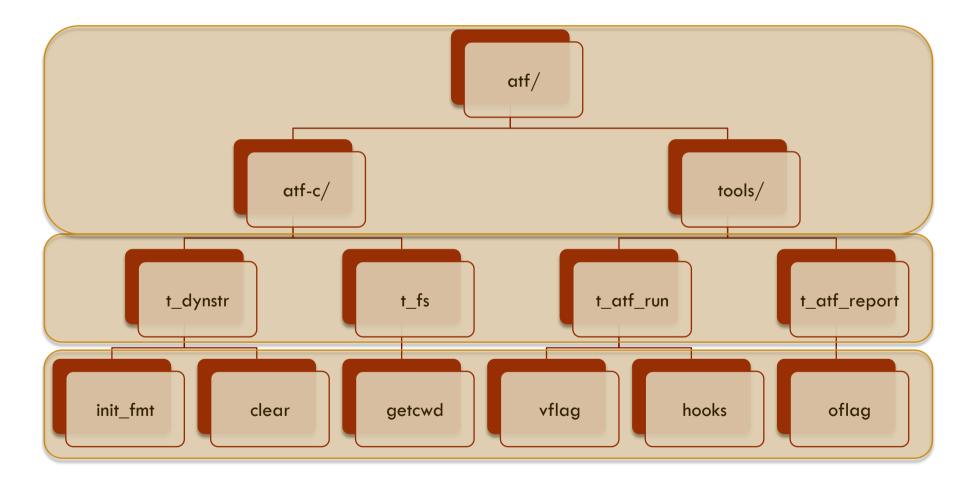
The beginnings of ATF



Major features of ATF

- □ C, C++ and POSIX shell libraries
- Interface consistency among tests
- □ No need for developer-only tools
 - Good bye dear (dear?) make(1)!
- Tests are installed

Test suite structure

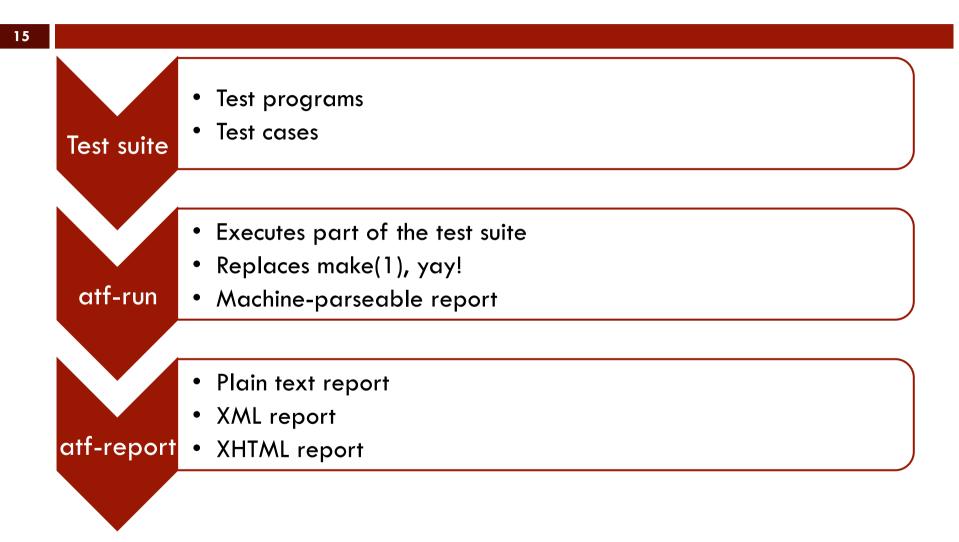


The ATF libraries

Provide support to write test cases/programs
In C, C++ and POSIX shell

- □ What do they provide?
 - Consistent interface to the end user
 - Test case isolation
 - And report the results!
- Easy to create libraries for new languages

The ATF tools



The new NetBSD tests

- NetBSD tests migrated to ATF
- Located in src/tests
- □ New installation set: tests.tgz
 - Installed into /usr/tests
- □ SoC 2008 project
 - Migrate src/regress to src/tests

Conclusions

- 17
- Automated tests are important
 - And actually running them periodically is too!
- □ ATF provides a generic solution
 - Shameless plug: use it to test your own software!
- □ Still work to do, though:
 - Kernel testing: how?
 - Distributed testing: SMOP?

References

- 18
- □ Web site
 - <u>http://www.NetBSD.org/~jmmv/atf/</u>
- Mailing list
 - <u>http://www.NetBSD.org/mailinglists/#atf-devel</u>
- Bleeding-edge development code

<u>http://mtn-host.prjek.net/viewmtn/atf/</u>

Latest release

0.5